

**Yee &  
Associates, P.C.**

4100 Alpha Road  
Suite 1100  
Dallas, Texas 75244

Main No. (972) 385-8777  
Facsimile (972) 385-7766

RECEIVED  
CENTRAL FAX CENTER

JUN 23 2005

## Facsimile Cover Sheet

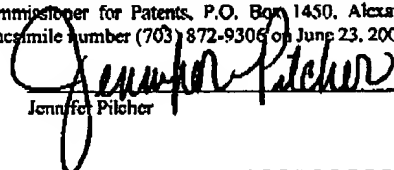
To: Commissioner for Patents for Examiner Hussein A. El Chanti Group Art Unit 2157	Facsimile No.: 703/872-9306
From: Jennifer Pilcher Legal Assistant to Brian Owens	No. of Pages Including Cover Sheet: 28
Message:  Transmitted herewith: <ul style="list-style-type: none"><li>• Transmittal Document; and</li><li>• Appeal Brief.</li></ul>	
Re: Application No.: 09/784,694 Attorney Docket No: RSW920010011US1	
Date: Thursday, June 23, 2005	
<b>Please contact us at (972) 385-8777 if you do not receive all pages indicated above or experience any difficulty in receiving this facsimile.</b>	<i>This Facsimile is intended only for the use of the addressee and, if the addressee is a client or their agent, contains privileged and confidential information. If you are not the intended recipient of this facsimile, you have received this facsimile inadvertently and in error. Any review, dissemination, distribution, or copying is strictly prohibited. If you received this facsimile in error, please notify us by telephone and return the facsimile to us immediately.</i>

**PLEASE CONFIRM RECEIPT OF THIS TRANSMISSION BY  
FAXING A CONFIRMATION TO 972-385-7766.**

## IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re application of: **Martin et al**Serial No.: **09/784,694**Filed: **February 15, 2001****For: Method and System for  
Specifying a Cache Policy for Caching  
Web Pages Which Include Dynamic  
Content****36736**PATENT TRADEMARK OFFICE  
CUSTOMER NUMBER

§ Group Art Unit: **2157**  
§  
§ Examiner: **El Chanti, Hussein A.**  
§  
§ Attorney Docket No.: **RSW920010011US1**  
§

<p><b>Certificate of Transmission Under 37 C.F.R. § 1.3(a)</b> I hereby certify this correspondence is being transmitted via facsimile to the Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450, facsimile number (703) 872-9306 on June 23, 2005. By:  Jennifer Pilcher</p>
---

**TRANSMITTAL DOCUMENT**

Commissioner for Patents  
P.O. Box 1450  
Alexandria, VA 22313-1450

Sir:  
ENCLOSED HEREWITH:

- Appeal Brief (37 C.F.R. 41.37).

A fee of \$500.00 is required for filing an Appeal Brief. Please charge this fee to IBM Corporation Deposit Account No. 09-0461. No additional fees are believed to be necessary. If, however, any additional fees are required, I authorize the Commissioner to charge these fees which may be required to IBM Corporation Deposit Account No. 09-0461. No extension of time is believed to be necessary. If, however, an extension of time is required, the extension is requested, and I authorize the Commissioner to charge any fees for this extension to IBM Corporation Deposit Account No. 09-0461.

Respectfully submitted,



Brian Owens

Registration No. 55,517

PATENT AGENT FOR APPLICANTS

Duke W. Yee

Registration No. 34,285

ATTORNEY FOR APPLICANTS

YEE &amp; ASSOCIATES, P.C.

P.O. Box 802333

Dallas, Texas 75380

(972) 385-8777

RECEIVED  
CENTRAL FAX CENTER

JUN 23 2005

PATENT

Docket No. RSW920010011US1

## IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re application of: **Martin et al.**Serial No. **09/784,694**Filed: **February 15, 2001**For: **Method and System for Specifying  
a Cache Policy for Caching Web Pages  
Which Include Dynamic Content**§  
§  
§  
§  
§  
§Group Art Unit: **2157**Examiner: **El Chanti, Hussein A.**Attorney Docket No.: **RSW920010011US1**Commissioner for Patents  
P.O. Box 1450  
Alexandria, VA 22313-1450Certificate of Transmission Under 37 C.F.R. § 1.8(a)I hereby certify this correspondence is being transmitted via  
facsimile to the Commissioner for Patents, P.O. Box 1450,  
Alexandria, VA 22313-1450, facsimile number (703) 872-9306  
on June 23, 2005.

By:

Jennifer Pilcher

## APPEAL BRIEF (37 C.F.R. 41.37)

This brief is in furtherance of the Notice of Appeal, filed in this case on April 25, 2005.

The fees required under § 41.20(B)(2), and any required petition for extension of time for filing this  
brief and fees therefore, are dealt with in the accompanying TRANSMITTAL OF APPEAL BRIEF.(Appeal Brief Page 1 of 26)  
Martin et al. - 09/784,694

**REAL PARTY IN INTEREST**

The real party in interest in this appeal is the following party: International Business Machines Corporation

**RELATED APPEALS AND INTERFERENCES**

With respect to other appeals or interferences that will directly affect, or be directly affected by, or have a bearing on the Board's decision in the pending appeal, there are no such appeals or interferences.

**STATUS OF CLAIMS**

**A. TOTAL NUMBER OF CLAIMS IN APPLICATION**

Claims in the application are: 1-42.

**B. STATUS OF ALL THE CLAIMS IN APPLICATION**

1. Claims canceled: NONE
2. Claims withdrawn from consideration but not canceled: NONE
3. Claims pending: 1-42
4. Claims allowed: NONE
5. Claims rejected: 1-42
6. Claims objected to: NONE

**C. CLAIMS ON APPEAL**

The claims on appeal are: 1-42

**STATUS OF AMENDMENTS**

A Response to the Final Office Action was filed on March 23, 2005; however, no amendments were made to the claims in the Response. Claims 1-42 on appeal herein are as finally rejected in the Final Office Action mailed January 25, 2005.

### **SUMMARY OF CLAIMED SUBJECT MATTER**

#### **A. Independent Claims 1, 15 and 29:**

The subject matter of claim 1 is directed toward a method, system, and computer program for specifying a cache policy for caching pages which include dynamic content. A system in which the invention may be implemented is illustrated in **Figure 4** and is described beginning on page 9, line 15 and extending to page 12, line 6. A user is permitted to request one of the pages to be displayed. See page 10, line 19-26. The page includes multiple fragments. See page 11, lines 20-28. An application is executed that includes multiple servlets, each of which is executed to present a different one of the multiple fragments. See page 6, lines 19-20. The servlets are unchanged by the caching policy. See page 12, lines 1-6. See page 12, lines 1-6. Each one of the servlets is executed to present a different one of the fragments. See page 6, lines 20-21. Caching of the page fragments is processed separately from the execution of the application and its servlets. See page 12, lines 1-3. One of multiple cache options is specified based on an update rate of content of one of the multiple servlets. The options include static caching, dynamic caching or no caching. See **Figure 5**, page 19, line 5 through page 20, line 17. Content that is updated dynamically is cached using static caching, dynamic caching, or no caching.

#### **B. Dependent Claims 6, 20, 34**

Claim 6, 20 and 34 are dependent on the method, system and computer program elements of claims 1, 15 and 29 respectively. Claim 6 further recites the steps of receiving a request to display one of the plurality of fragments (see page 9, lines 17-21, **step 802 Figure 8**), determining one of plurality of servlets associated with one of the plurality of fragments (see page 10, lines 7-26 and page 11, lines 14-28, and executing one of the plurality of servlets wherein the execution of one of the plurality of servlets generates a displayable output (see page 24 line 28 through page 25 line 6, **steps 822, 824 and 826 in Figure 8**).



**GROUND OF REJECTION TO BE REVIEWED ON APPEAL**

**The grounds of rejection on appeal are as follows:**

- I. Claims 1-42 are rejected under 35 U.S.C. §103(a) as being allegedly unpatentable over Hawcs (U.S. Patent No. 6,094,662).

## ARGUMENT

### **I. 35 U.S.C. §103(a), Alleged Obviousness Claims 1-42.**

The Final Office Action rejects claims 1-42 under 35 U.S.C. § 103(a) as being unpatentable over *Hawes*. (U.S. Patent No. 6,094,662) (hereinafter "*Hawes*"). This rejection should be reversed.

#### **IA. Claims 1-5, 15-19, 29-33**

*Hawes* teaches an apparatus and method for loading and reloading HTML pages having cacheable and non-cacheable portions. The HTML portions of the page are marked as non-cacheable which allows the web page to be retrieved without retrieving cacheable graphics images. A refresh function allows a time stamp of non-cached portions to be compared with the time stamps of the web server so that when the web server has a more current time stamp, a browser displaying the web page can be updated with the non-cached portion. Additionally, a timer or refresh button on the web page or browser may be employed to update the non-cached portion of the web page.

In rejecting the claims as being obvious over *Hawes*, the Examiner states the following:

As to claims 1, 15 and 29. *Hawes* teaches a method, system, and program respectively in a data processing system for specifying a cache policy for caching pages which include dynamic content, said method, system and program comprising the steps of:

- permitting a user to request one of said pages to be displayed, said one of said pages including a plurality of fragments (see col. 4 lines 45-47, user retrieves web pages);

- executing an application which includes a plurality of HTML portions, each one of said plurality of HTML portions being executed to present a different one of said plurality of fragments, each one of said plurality of HTML portions being unchanged by said caching policy (see col. 4 lines 48-52, browser separates cacheable and non-cacheable portions); and

- processing caching of said one of said pages separately from said application; and

- specifying one of a plurality of different caching options for one of said plurality of HTML portions based on an update rate of content of said one of said plurality of HTML portions, said plurality of different caching options including either static caching, dynamic caching or no caching content that is updated dynamically being cached either static caching, dynamic caching or no caching

(see col. 5 lines 13-36, browser dynamically and periodically updates the cacheable portion).

*Hawes* does not explicitly teach the HTML portions are servlets. Official Notice is taken as evident by Microsoft computer Dictionary 5<sup>th</sup> edition that it would have been obvious for one of the ordinary skill in the art at the time of the invention to modify *Hawes* by using servlets because doing so would make the browser execute quickly and thereby reducing system overhead.

Final Office Action dated March 25, 2005 pages 2 and 3.

Claim 1 is a representative claim in this group and states as follows:

1. A method in a data processing system for specifying a cache policy for caching pages which include dynamic content, said method comprising the steps of:
  - permitting a user to request one of said pages to be displayed, said one of said pages including a plurality of fragments;
  - executing an application which includes a plurality of servlets, each one of said plurality of servlets being executed to present a different one of said plurality of fragments, each one of said plurality of servlets being unchanged by said caching policy;
  - processing caching of said one of said pages separately from said application; and
  - specifying one of a plurality of different caching options for one of said plurality of servlets based on an update rate of content of said one of said plurality of servlets, said plurality of different caching options including either static caching, dynamic caching or no caching, content that is updated dynamically being cached using either static caching, dynamic caching, or no caching.

Claim 1 recites an application that includes multiple servlets. Each servlet is executed to presents a different one of the multiple fragments of a page. *Hawes* does not suggest using servlets and particularly does not suggest processing caching of the page separate from the application. Page 12, lines 1-6.

The Examiner alleges this argument is ineffective by stating the following in the Advisory Action:

Applicant's arguments were fully considered but are not persuasive. In remarks, applicant argues A) *Hawes* does not disclose processing caching pages separately from the application.

In response *Hawes* teaches a system and method for caching webpages by checking HTML portions and determining whether the portion are "cacheable" or "non-cacheable". The determination as to whether the portions are "cacheable" or "non-cacheable" are made using the portions characteristics and not based on the application that requested the application (see col. 6, lines 15-42) and therefore

Hawes meets the scope of the claimed limitation "processing caching of said one of said pages separately from said application".

Advisory Action dated April 4, 2005 cover page.

This statement is contradicted by specification and claims of *Hawes*. Particularly, the single apparatus claim indicative of a "system" in *Hawes* that claims in part:

3. An apparatus that updates a document retrieved from a node of a distributed network to a client, comprising:

a browser that retrieves the document from the node over the distributed network and separates the document into at least one cacheable portion and at least one non-cacheable portion, the at least one cacheable portion stored in-cache memory of the client and the at least one non-cacheable portion stored in non-cache memory of the client, the browser comprising:

a refresh status button that compares each at least one non-cacheable portion of the document stored in the non-cache memory with a corresponding non-cacheable portion of the document stored at the node;

wherein, for each non-cacheable portion of the document stored in the non-cache memory that is different from the corresponding non-cacheable portion of the document stored at the node, the browser retrieves the corresponding non-cacheable portion of the document stored at the node and stores the retrieved non-cacheable portion in the non-cache memory in place of the corresponding non-cacheable portion previously stored in the non-cacheable memory." (emphasis added)

*Hawes* col. 8, lines 25-48.

Clearly the application or browser in this case is processing the cacheable and non-cacheable portions of the page by separating, retrieving and storing them. *Hawes* further notes in that:

In the preferred embodiment, the browser 180 accesses the web page URL, and compares the non-cached portion with the corresponding portion of the web page 212. If a change is detected, the browser 180 requests the non-cacheable portion of the web page 212 to be downloaded to the client 110 and displayed on display 140. While retrieving the non-cacheable portion of the web page 212 from the web site 210, the browser retrieves the cached portion of the web page 212 from cache 184 for displaying on display. 140

See *Hawes* col. 5, lines 13-37.

The "portions characteristics" described by the examiner are used by the browser in processing the caching. The portions characteristics are not used independent of the browser to process the cached elements as suggested.